

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2003-022250

(43)Date of publication of application : 24.01.2003

(51)Int.Cl.

G06F 15/00  
G06F 13/00

(21)Application number : 2001-207912

(71)Applicant : NEC CORP

(22)Date of filing : 09.07.2001

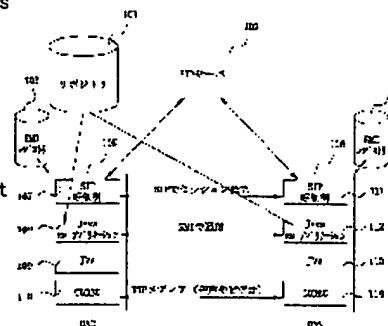
(72)Inventor : ARAO SHINYA

## (54) SESSION ESTABLISHING METHOD

## (57)Abstract:

PROBLEM TO BE SOLVED: To provide a session establishing method for applying an SIP(Session Initiation Protocol) to the establishment (session establishment) of communication between Java RMI distributed objects in order to realize the various applications of Peer-to-Peer communication.

SOLUTION: The session establishing method comprises a procedure for allowing a first information processor to start an application and to make applications for a session to a second information processor by using an SIP, a procedure for allowing the second information processor to permit the session to the first information processor by using the SIP, a procedure for allowing the second information processor to start the application, and a procedure for allowing the first information processor and the second information processor to establish the session between the started applications. The parameter of the application is included in the payload in the procedure for making applications for or permitting the session by using the SIP.



## LEGAL STATUS

[Date of request for examination]

12.06.2002

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号  
特開2003-22250  
(P2003-22250A)

(43) 公開日 平成15年1月24日 (2003.1.24)

(51) Int.Cl. <sup>7</sup>	識別記号	F I	ターミナル <sup>7</sup> (参考)
G 0 6 F 15/00	3 1 0	G 0 6 F 15/00	3 1 0 D 5 B 0 8 5
13/00	3 5 3	13/00	3 5 3 C 5 B 0 8 9

審査請求 有 請求項の数 7 O L (全 10 頁)

(21) 出願番号 特願2001-207912(P2001-207912)

(22) 出願日 平成13年7月9日(2001.7.9)

(71) 出願人 000004237

日本電気株式会社  
東京都港区芝五丁目7番1号

(72) 発明者 荒生 慎也

東京都港区芝五丁目7番1号 日本電気株式会社内

(74) 代理人 100095740

弁理士 関口 宗昭

Fターム(参考) 5B085 A002 B000

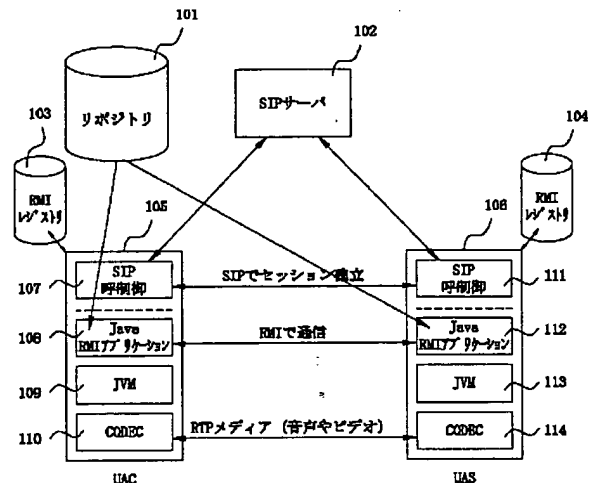
5B089 GA07 GB01 HA01

(54) 【発明の名称】 セッション確立方法

(57) 【要約】

【課題】 Peer-to-Peer通信の色々なアプリケーションを実現させるための、SIPをJava RMI分散オブジェクト間の通信の確立(セッションの確立)を行うことへ適用した、セッション確立方法を提供する。

【解決手段】 第1の情報処理装置がアプリケーションを起動して、SIP(Session Initiation Protocol)を用いて第2の情報処理装置に対してセッションの申込を行う手順と、第2の情報処理装置がSIPを用いて第1の情報処理装置に対してセッションの許可を行う手順と、第2の情報処理装置がアプリケーションを起動する手順と、第1の情報処理装置と第2の情報処理装置とにおいて起動されているアプリケーション間でセッションの確立を行う手順とを有し、SIPを用いたセッションの申込または許可を行う手順において、ペイロードにアプリケーションのパラメータを含ませる。



## 【特許請求の範囲】

【請求項1】第1の情報処理装置がアプリケーションを起動して、SIP (Session Initiation Protocol) を用いて第2の情報処理装置に対してセッションの申込を行う手順と、第2の情報処理装置がSIPを用いて前記第1の情報処理装置に対してセッションの許可を行う手順と、前記第2の情報処理装置が前記アプリケーションと同等のアプリケーションを起動する手順と、前記第1の情報処理装置と前記第2の情報処理装置とにおいて起動されているアプリケーション間でセッションの確立を行う手順とを有することを特徴とするセッション確立方法。

【請求項2】前記第1の情報処理装置および/または前記第2の情報処理装置は、起動するアプリケーションが保存されたリポジトリからダウンロードして起動することを特徴とする請求項1に記載されたセッション確立方法。

【請求項3】前記第1の情報処理装置は起動したアプリケーションのリファレンスを第1のレジストリに登録し、前記第2の情報処理装置は起動したアプリケーションのリファレンスを第2のレジストリに登録し、互いのリファレンスを取得しあうことを特徴とする請求項1または請求項2に記載されたセッション確立方法。

【請求項4】前記SIPを用いたセッションの申込または許可を行う手順で、ペイロードに前記アプリケーションのパラメータを含ませることを特徴とする請求項1乃至請求項3の何れかに記載されたセッション確立方法。

【請求項5】前記パラメータには、前記第1のレジストリまたは前記第2のレジストリのURLが記述されていることを特徴とする請求項4に記載されたセッション確立方法。

【請求項6】前記パラメータには、前記リポジトリのURLが記述されていることを特徴とする請求項4または請求項5に記載されたセッション確立方法。

【請求項7】前記パラメータはXML形式で記述されていることを特徴とする請求項4乃至請求項6の何れかに記載されたセッション確立方法。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】本発明は、Peer-to-Peerアプリケーションのセッション確立方法に関する。

## 【0002】

【従来の技術】Javaは色々なオペレーティングシステムやプラットフォーム上に広く普及しており、SIP (Session Initiation Protocol. IETF RFC2543) もPeer-to-Peerアプリケーションのセッション確立のためのプロトコルとして普及してきている。SIPはペイロードにSDP以外のデータを持つことも出来、分散オブジェクト間の通信の確立 (セッションの確立) を行うこ

とも利用できる。しかし、Javaによる分散オブジェクトを使った、インタラクティブなアプリケーションは使われていない。

【0003】RMIオブジェクトのアドレスであるRMI URLを保持するNaming Serverが一般的に広く普及していないために、どんなRMIオブジェクトがどのPCやサーバで利用可能か知ることが出来なかった。また、個人が所有するPCはパワーオフしたり、ネットワークへのケーブルをPCから抜くために、頻繁にネットワークとの接続が切れるので、PCのような端末上でRMIオブジェクトが常に処理受付状態で稼働している状況はなかなか実現しにくかった。そのため、Peer-to-peer通信のJavaアプリケーションは、これまでうまく実現できなかった。

## 【0004】

【発明が解決しようとする課題】従って本願発明は、Peer-to-Peer通信の色々なアプリケーションを実現させるための、SIPをJava RMI分散オブジェクト間の通信の確立 (セッションの確立) を行うことへ適用した、セッション確立方法を提供することを課題とする。

## 【0005】

【課題を解決するための手段】前記課題を解決するための本願発明のセッション確立方法は、第1の情報処理装置がアプリケーションを起動して、SIP (Session Initiation Protocol) を用いて第2の情報処理装置に対してセッションの申込を行う手順と、第2の情報処理装置がSIPを用いて前記第1の情報処理装置に対してセッションの許可を行う手順と、前記第2の情報処理装置が前記アプリケーションと同等のアプリケーションを起動する手順と、前記第1の情報処理装置と前記第2の情報処理装置とにおいて起動されているアプリケーション間でセッションの確立を行う手順とを有することを特徴とする。

【0006】これにより、SIPをサポートする情報処理装置では、発呼者が任意のRMIアプリケーションを自分の端末と、複数の着信者の端末上で起動し、それらの間のPeer-to-Peer通信を実現できる。これによって、遠隔教育、地域コミュニティーでのだんらんやネットワークゲームなど、色々なPeer-to-Peer通信サービスを展開できるようになる。

【0007】また、前記課題を解決するための本願発明のセッション確立方法は、前記第1の情報処理装置および/または前記第2の情報処理装置は、起動するアプリケーションが保存されたリポジトリからダウンロードして起動することを特徴とする。

【0008】これにより、第1の情報処理装置と第2の情報処理装置が、該当するアプリケーションを保有していない場合にも、リポジトリからアプリケーションをダウンロードして起動を行うことができる。

【0009】また、前記課題を解決するための本願発明のセッション確立方法は、前記第1の情報処理装置は起動したアプリケーションのリファレンスを第1のレジストリに登録し、前記第2の情報処理装置は起動したアプリケーションのリファレンスを第2のレジストリに登録し、互いのリファレンスを取得しあうことを特徴とする。

【0010】第1及び第2の情報処理装置が、アプリケーションのリファレンスをレジストリに登録して、互いのリファレンスを取得しあうことにより、ローカルなオブジェクトのJava APIを利用するのと同じ要領で、相手リモートオブジェクトを利用できる。

【0011】また、前記課題を解決するための本願発明のセッション確立方法は、前記SIPを用いたセッションの申込または許可を行う手順で、ペイロードに前記アプリケーションのパラメータを含ませることを特徴とする。

【0012】SIPのペイロードに、アプリケーションのパラメータを含ませることにより、第1の情報処理装置と第2の情報処理装置それぞれにおけるJavaオブジェクト間のRMI通信の確立に必要なパラメータを、第1の情報処理装置が第2の情報処理装置へ送ることができる。

【0013】また、前記課題を解決するための本願発明のセッション確立方法は、前記パラメータには、前記第1のレジストリまたは前記第2のレジストリのURLが記述されていることを特徴とする。

【0014】パラメータにレジストリのURLが記述されていることにより、両情報処理装置が、互いのアプリケーションのリファレンスを互いに取得することが可能になる。

【0015】また、前記課題を解決するための本願発明のセッション確立方法は、前記パラメータには、前記リポジトリのURLが記述されていることを特徴とする。

【0016】パラメータにリポジトリのURLが記述されていることにより、両情報処理装置において起動すべきアプリケーションを、確実にダウンロードすることができる。

【0017】また、前記課題を解決するための本願発明のセッション確立方法は、前記パラメータはXML形式で記述されていることを特徴とする。を特徴とする。

【0018】パラメータをXML形式で記述することで、アプリケーション毎に異なる仕様のパラメータに対しても、柔軟に対応して必要な内容を記述することが可能となる。

【0019】

【実施の形態】図1は実施の形態としての、本願発明のSIPによるJava RMIセッション確立方法を用いたシステムの基本構成図である。SIPでは、マルチメディア通信端末105はUAC (User Agen

t Client)、106はUAS (User Agent Server)と呼ばれる。UAC105はUAS106に対しSIPを使ってマルチメディアセッションの確立を行う。SIPメッセージのペイロードSDP (Session Description Protocol. IETF RFC2327) にRTP (Realtime Transport Protocol. IETF RFC1889) のメディア属性を記述することで、複数のマルチメディア端末間の音声やビデオ通信を確立することが出来る。

【0020】本実施の形態では、SIPのペイロードとしてJava RMI (Remote Method Invocation) アプリケーションに関係したパラメータをXMLで記述し、複数のマルチメディア端末上で、指定されたJavaオブジェクトのインスタンス生成と起動を行い、その間でRMIを使ったPeer-to-Peerの通信の確立を行うことを特徴とする。ここでは、自端末から他端末上のJavaオブジェクトをRMIを使って利用する場合、他端末上のJavaオブジェクトをリモートオブジェクトと称する。また、本発明で言及されるJavaプログラムは全てRMIを利用するため、以後RMIアプリケーションと称することにする。

【0021】図1においてSIPサーバ102があるが、これはSIPメッセージのルーティングに使われる。例えば、UAC (105) がUAS (106) へSIPメッセージを送るとき、SIPサーバ102を経由して送る。これは本発明には本質的な要素ではない。同様に、CODEC110、及び114はRTPメディアである音声やビデオの符号化/復号化に使われるのみで、本発明において本質的な要素ではない。109および113はJVM (Java Virtual Machine) であり、Javaが実行可能なオペレーティングシステムやプラットフォーム上で必ず提供されているものである。

【0022】図1において、UAC (105) のSIP呼制御機能107はUAS (106) のSIP呼制御機能111とSIPのシグナリングメッセージを交換することで、UACとUAS間のセッションを確立する。このときSIPメッセージの中のペイロードとして、RTPメディアのセッションを確立するためにSDPが運ばれ、リモートオブジェクト間のセッションを確立するためにXMLで記述された関連パラメータが運ばれる。具体的には、UACがSIPでセッションを確立するとき、UAS106において起動してほしいJavaオブジェクト (RMIアプリケーション) 名とそれに関係するパラメータをXMLの形で記述し、それをSIPメッセージのペイロードでUAS106へ送る。リポジトリ101に、JavaオブジェクトがJAR (Java Archive File) の形式で保存されており、

UAC (105) と UAS (111) は、HTTP URL で指定された特定の JAR ファイルを HTTP (IETF RFC 2616) を使ってダウンロードする事が出来る。

【0023】UAC (105) は、SIP セッション確立前に自端末において起動された RMI アプリケーション (108) を RMI レジストリ 103 へ名前をつけて登録する。登録後、UAS (106) は、UAC (105) において起動された Java オブジェクトの RMI リファレンスを、その Java オブジェクトに相当する RMI URL を指定することで取得できるようになる。同様に、UAS (106) の SIP 呼制御機能 (111) は、SIP セッション確立途中に自端末において起動された Java オブジェクトを RMI レジストリ 104 へ名前をつけて登録する。登録後、UAC (105) は、UAS (106) において起動された RMI アプリケーション (112) の RMI リファレンスを、その Java オブジェクトに相当する RMI URL を指定することで取得できるようになる。

【0024】図2は、RMI 分散オブジェクト間のセッション確立時における本実施の形態の動作を示した図である。発呼者側マルチメディア端末 (SIP UAC) は、ある任意の RMI アプリケーション 201 を起動する。201 は SUN Microsystems が JAIN として標準化を行っているようなある API を使って、SIP セッション起動 202 を行う。この時、UAS 側マルチメディア端末でどの RMI アプリケーションが起動されるべきか、また、RMI レジストリ 103 に登録した RMI URL を SIP 呼制御 107 へ知らせる。

【0025】UAC は INVITE メッセージ (203) を UAS へ送る。UAS は直ちに 180 Ringing (204) を UAC へ返し、INVITE のペイロードに XML で指定された Java オブジェクトのインスタンス生成を行う (208)。ただし、このインスタンスが生成されても良い事を UAS のユーザに事前に確認して起動許可を受ける (206 および 207)。この確認には、発呼者の名前として INVITE メッセージの中の From ヘッダーフィールドの値を表示するとともに、INVITE メッセージ内の Subject ヘッダーフィールドに示されたサブジェクト名も表示する。それらヘッダーフィールドの INVITE メッセージ例を図4に示す。図3のウインドウ (301) がその確認画面である。UAS 側ユーザはこのウインドウの Yes ボタン (302) を押すことで、指定された Java オブジェクトが起動される事を許可し、このセッションを継続する。もし、許可しない場合は、No ボタン (303) を押す。

【0026】図4において、実際の SIP メッセージの例を示した。ここで、INVITE メッセージの中の 4

04 が SDP ペイロードで、407 が本発明の特徴である XML ペイロードである。この中において、405 で MIME (IETF RFC 2045) タイプ名である "application/x-java-rmiparam" が記述されている。"x-" で始まる MIME タイプ名は拡張仕様であり、IANA での正式登録なしに使用することが出来る。したがって、MIME タイプ名に関しては、そのマルチメディア端末が解釈できる任意の名前をとる事が出来る。また、XML でのパラメータ記述に必要なシンタックスやセマンティクスは XML での DTD (Document Type Definition) で任意に定義することが出来る。

【0027】図4において、XML 記述 407 の <jarurl>http://xyz\_school.co.jp/lessons/unit25.jar</jarurl> と <rmiurl>rmi://greg.xyz\_school.co.jp/unit25</rmiurl> の2つが JAR ファイルの HTTP URL と RMI URL を示すパラメータに相当する。<jarurl>, </jarurl>, <rmiurl>, </rmiurl> 等の XML タグは DTD で任意に定義できる。なお図4の 401, 402, 406 はこのペイロードの中身に応じて変わる数字である。401 は 404 と 405 から構成される2つの MIME オブジェクトのトータルのバイト長で、実際には数字が入る。402 は 404 の SDP 部分のバイト長、404 は 407 の XML 部分のバイト長がそれぞれ数字で入る。404 の SDP の MIME オブジェクトと 407 の XML の MIME オブジェクトは、この例では "boundary" という境界で区切られる。

【0028】XML 記述 407 において指定されたパラメータによって、UAS はレジストリ 101 から、HTTP URL である "http://xyz\_school.co.jp/lessons/unit25.jar" にある unit25.jar という名前のファイルを HTTP でダウンロードし、それを起動する。Java のプログラムを起動するには main (...) というメソッド名がある Java クラスの名前を事前に知っておく必要があるが、どの Java クラスに main (...) メソッドが含まれるかは JAR ファイル内の Manifest に記述されているので、UAS はそれを参照する。

【0029】図2において、UAS における指定された RMI アプリケーションの起動後、図4の INVITE メッセージの XML 記述 407 内のパラメータ <rmiurl>rmi://greg.xyz\_school.co.jp/unit25</rmiurl> で指定された RMI URL である "rmi://greg.xyz\_school.co.jp/unit25" を使って、UAC における RMI アプリケーション

のリファレンスを取得する(209)。また、ここではPeer-to-Peerの通信アプリケーションを前提としているため、この時、逆に、UASで起動されているRMIアプリケーションをUACがRMIを使って利用するケースも考えられる。

【0030】これは図1において、RMIアプリケーション112がRMIレジストリ104へ自分自身のRMI URLを登録し、それをRMIアプリケーション108へ知らせる事で、RMIアプリケーション108がRMIアプリケーション112のリファレンスを取得することが可能となる。この処理はRMIアプリケーションのプログラムアルゴリズム次第となる。また、これに関しては、まだInternet Draftの段階であるが、RFC2543の改版であるRFC2543-bis内のSIPメッセージ183 Session Progress を使うこともできる。INVITE内のXMLペイロードと同様な方法で183 Session Progress のペイロードとして上記RMI URLをUASからUACへ運ぶ事ができる。

【0031】図2において、UACとUASの間のRMIアプリケーション間のセッション209の確立が成功した場合、Javaアプリケーション(RMIアプリケーション)は210において、準備OKであることをUASのSIP呼制御111へ知らせる。SIP呼制御は200 OK(211)をUACへ返し、UACはACK(212)を返す。これによって、RMIを含むマルチメディアセッションがSIPによって確立される。この時、図3にあるウィンドウ304が、UAS側ユーザへ示され、RMI含むマルチメディアセッションが確立したことを知らせる。

【0032】このセッションは、分散オブジェクト間のセッション確立だけでなく、音声やビデオなどのRTPメディアのセッションも確立できる。SIPはMIME Multipartに対応するため、複数のMIME タイプをペイロードに保持し、複数のMIMEオブジェクトを運べるからである。例として図4にMIMEMultipartで記述されたペイロードを示す。この例で、SDP記述403では、ITU-T G. 711 μlawでの音声符号化方式によるRTPメディアと、ITU-T H. 261によるビデオ符号化方式によるRTPメディアが指定されている。したがって、この例では、RMIアプリケーション間のPeer-to-Peer通信と、音声、ビデオのRTPメディアによる通信が成立している事になる。

【0033】以上の実施の形態を実際のアプリケーション利用に置き換えて、外国語遠隔レッスンをとりあげて図5を用いて説明する。ここでは、講師と2人の生徒のマルチメディア端末間で、遠隔英会話教育を行うためのセッションを確立する場合を述べる。

【0034】外国語講師であるグレッグ先生が、あるレ

ッソンのRMIアプリケーション(Javaプログラム)を起動する。この例では、ユニット25のレッスンを行うためのプログラムが起動される。必要なJavaクラスはJARファイルとして、リポジトリ501からダウンロード(302)される。この語学学習プログラム(504)は、SIPを使って、生徒A(太郎)及び生徒B(花子)とのセッションを確立する。

【0035】図5において、SIPとJavaをサポートする2つのマルチメディア端末515、516が示される。515がUAC、516がUASに相当する。これらマルチメディア端末は、それぞれの顔と音声をお互いに送るため、ビデオカメラとマイクが搭載されている。また、内部には、SIP呼制御処理、JVM(Java Virtual Machine)と音声とビデオの符号化/復号化に使われるCODECが装備されている。この外国語スクールで使用する教材は複数のユニット(単位)で構成されるとし、リポジトリ501は、それぞれのユニットをJARファイルの形で保持している。

【0036】英語学習プログラムのユニット25は、UACでは504、UASでは505としてJVM上で稼動している。これらはRMIによってお互いに通信する(506)。CODECにより、音声とビデオのRTPメディア(507、508)が符号化/復号化される。509はこの英会話レッスン、ユニット25のJavaプログラムのGUIである。510はグレッグ先生が英文テキストの一部を一時的に隠しており、511はクイズ形式のレッスンを行うためのものである。512にはグレッグ先生の顔が映し出され、513には生徒A、514には生徒Bの顔が映し出される。

【0037】図1、図2で示した動作の仕組みで本実施例が実現される。また、図3のウィンドウ301と304がマルチメディア端末516のディスプレイ上に、すなわち、ユーザへ示される。図4で示されるINVITEメッセージが、図5におけるマルチメディア端末515からマルチメディア端末516へ送出される。RMIアプリケーションはリポジトリ501にJARファイルで保持されている。図4のXML記述407にあるように、HTTP URLである"http://xyz\_school.co.jp/lessons/unit25.jar"から、マルチメディア端末515、516はユニット25のプログラムをJARファイルとしてダウンロードできる。マルチメディア端末516は、図4のXML記述407にあるように、RMIURLである"rmi://xyz\_school.co.jp/unit25"のアドレスにより、マルチメディア端末516のRMIアプリケーション505は、マルチメディア端末515上のRMIアプリケーション504のリファレンスを取得する。

【0038】実際にグレッグ先生と生徒A(太郎)、生

徒B（花子）が使用するユニット25のJavaプログラムのGUI（509）を図5右上に示す。このGUIはグレッグ先生、生徒Aと生徒Bのそれぞれのマルチメディア端末において表示される。これにおいて、GUIの中のウインドウ512にはグレッグ先生の顔が、ウインドウ513には生徒A（太郎）の顔が、ウインドウ514には生徒B（花子）の顔が表示される。これらはマルチメディア端末に搭載されているテレビカメラでとらえられた画像をRTPメディアとして送出しているものである。また、マルチメディア端末のマイクからとらえられたグレッグ先生と生徒A、生徒Bの声も、RTPメディアとして送出される。510はグレッグ先生が英文テキストのある一部を隠した状態で、511は3択式のクイズでBilly Joelの項をどちらかの生徒がクリックした例である。

【0039】生徒Aへのセッション確立と生徒Bへのセッション確立は同時に行える。SIPでは別の呼として扱われ、生徒AのRMIオブジェクトと生徒BのRMIオブジェクトは別々のRMI URLで識別されるからである。ただし、グレッグ先生のRMIオブジェクトは他の2つのRMIオブジェクトと通信していることを認識している必要がある。また、音声やビデオのRMIメディアを3者で共用するためには、IPマルチキャストか、もしくはグレッグ先生のマルチメディア端末515がRTPミキサーとして機能させる事が考えられる。

【0040】Peer-to-peer通信のJavaアプリケーションの利点は、図5の510に示されるように、グレッグ先生が英文テキストの任意の場所を隠し、そのGUIの同じ状態を先生と生徒A、生徒Bが共有している点。また511に示されるように、クイズ形式のインタラクティブな教材も実現できるところにある。遠隔教育でありながら、一つの机を囲んで、同じ教材を使いながら、先生と複数の生徒が英会話レッスンをしている状況がオンラインで実現できる。なお、ここでは外国語遠隔レッスンの実施例を取り上げたが、ネットワークゲーム等の他の色々なアプリケーションへ応用も可能である。

#### 【0041】

【発明の効果】SIPをサポートするマルチメディア端末では、発呼者が任意のRMIアプリケーションを自分の端末と、複数の着信者の端末上で起動し、それらの間のPeer-to-Peer通信を実現できる。これによって、遠隔教育、地域コミュニティーでの団欒やネットワークゲームなど、色々なPeer-to-Peer通信サービスを展開できるようになる。

【0042】マルチメディア端末は相手マルチメディア端末で起動しているリモートオブジェクトのリファレンスを取得することで、ローカルなオブジェクトのJava APIを利用するのと同じ要領で、相手リモートオブジェクトを利用できる。また、SIPでのセッション確

立時に、UACがUASへどのJavaオブジェクトを起動するのかを指定できるのと、UACとUASそれぞれにおけるJavaオブジェクト間のRMI通信の確立に必要なパラメータをUACがUASへ送ることができる。

【0043】SIPメッセージのペイロードSDPにRTPのメディア属性を記述することで、複数のマルチメディア端末間の音声やビデオ通信を確立することが出来る。

#### 【図面の簡単な説明】

【図1】実施の形態のシステム基本構成を示した図

【図2】RMI分散オブジェクト間のセッション確立手順を示した図

【図3】RMIアプリケーション起動確認ウィンドウとセッション確立ウィンドウの例

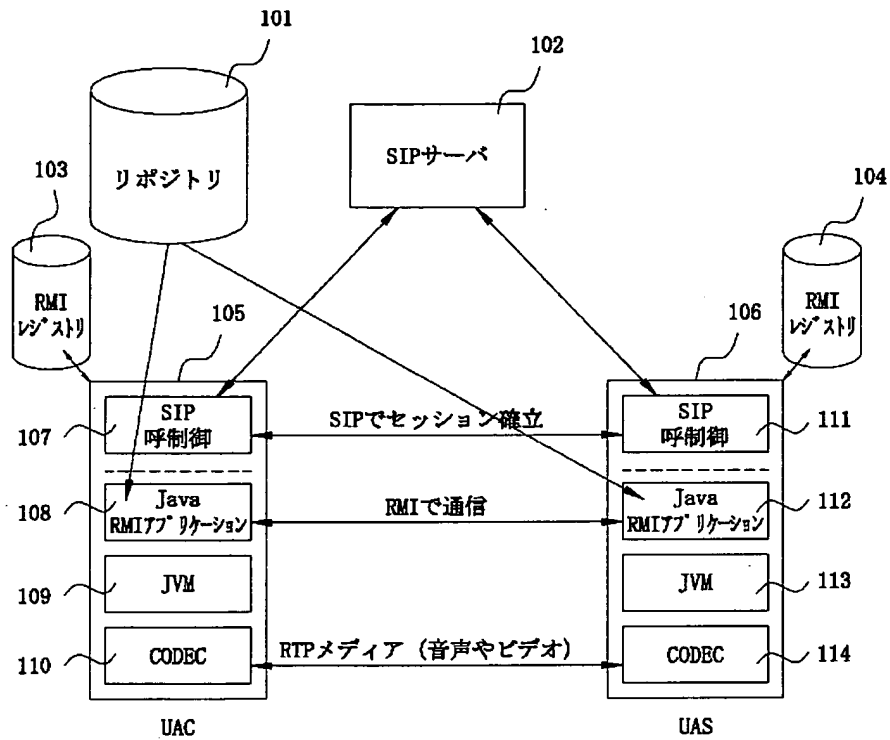
【図4】SIPに含まれるINVITEメッセージの例

【図5】遠隔英会話教育を例としたシステム構成を示した図

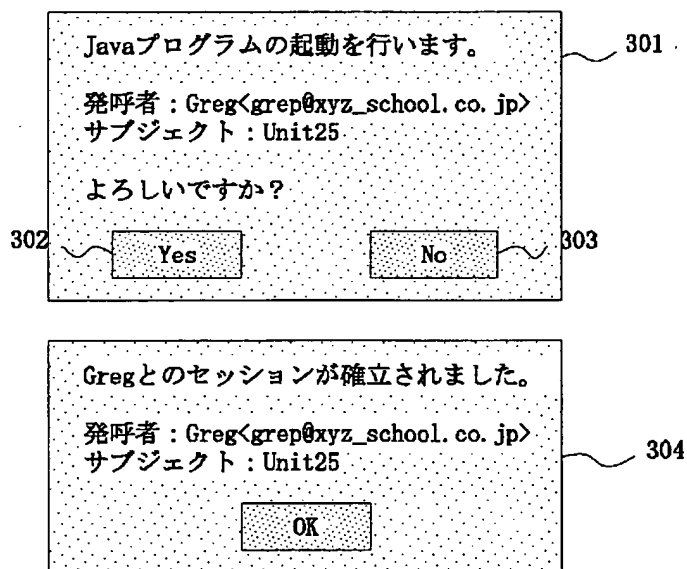
#### 【符号の説明】

101、501…リポジトリ  
 102…SIPサーバ  
 103、104…RMIレジストリ  
 105、515…UAC  
 106、516…UAS  
 107、111…SIP呼び制御  
 108、112…RMIアプリケーション  
 109、113…Java Virtual Machine  
 110、114…CODEC  
 201、208…RMIアプリケーション起動  
 202…SIPセッション起動  
 203…INVITEメッセージ  
 204…180 Ringing  
 206…起動確認  
 207…起動許可  
 209…セッション確立  
 210、213…準備OK  
 211…200 OK  
 212…ACK  
 301、302、303…起動確認ウィンドウ  
 304…セッション確立ウィンドウ  
 401、402、403、405、406…パラメータ  
 404…SDPペイロード  
 407…XMLペイロード  
 504、505…英語学習プログラム  
 507…音声  
 508…映像  
 509、510、511、512、513、514…GUI

【図1】

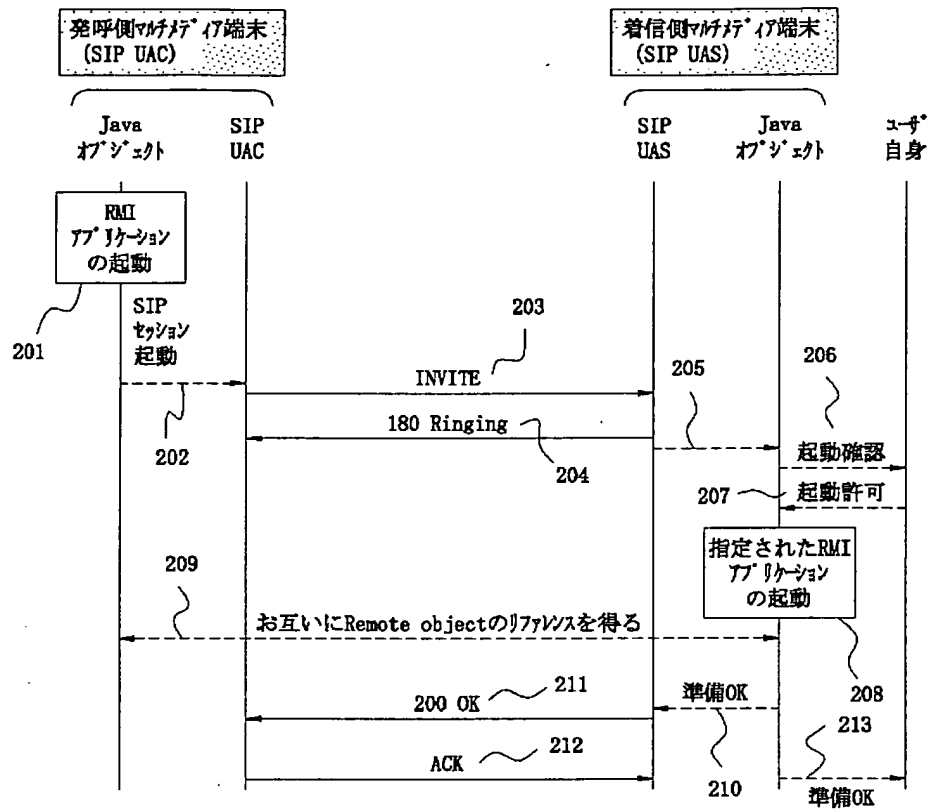


【図3】





【図2】



## 【図4】

```

INVITE sip:hanako@abc_isp.ne.jp SIP/2.0
Via: SIP/2.0/UDP pc123.xyz_school.co.jp:5060
From: Greg <sip:greg@xyz_school.co.jp>
To: Hanako <sip:hanako@abc_isp.ne.jp>
Call-ID: 20010331@pc123.xyz_school.co.jp
CSeq: 1 INVITE
Contact: sip:greg@pc123.xyz_school.co.jp
Subject: Unit 25
Content-Type: multipart/mixed; boundary="--boundary"
Content-Length: XXX ~ 401

--boundary
Content-Type: application/sdp
Content-Length: XXX ~ 402

v=0
o=greg 29739 7272939 IN IP4 126.5.4.3
c=IN IP4 135.180.130.88
m=audio 492170 RTP/AVP 0
m=video 3227 RTP/AVP 31
a=rtpmap:31 LPC

```

} 404

```

--boundary
Content-Type: application/x-java-rmiparam ~ 405
Content-Length: XXX ~ 406

<?xml version="1.0"?>
<!DOCTYPE PUBLIC "-//IETF//DTD RFCxxxx PMIPARAM 1.0//EN"
  "rmiparam.dtd">
<rmiparam>
  <jarurl>http://xyz_school.co.jp/lessons/unit25.jar</jarurl>
  <rmiurl>rmi://greg.xyz_school.co.jp/unit25</rmiurl>
</rmiparam>

```

} 407

```

--boundary

```

【図5】

